ISSN 0112-3351

BEEBLET

CONTENTS

Editorial 2
From Your President's Desk 3
Cross Hatch Generator 5
Lotto Number Generator 6
BigText 8
Classifieds 9, 28, 33 and 39
Saving Master Red Keys 10
Double Height Demonstration 17
Pattern - a one liner 19
Zoom Pixel Editor 20
Micro Interface (Advt) 26
Crashing Waves 28
Masterfile II - a review 29
HandiBags v Discs 31

Local User Group News

Wellington 32
Hawkes Bay 33
SelfTest 34
Service Agents and Dealers 39
Classifieds 9, 28, 33 and 39
Membership 39
Local User Group Meetings, Contacts 40
Beeblet - Contributions and Deadlines 40

Volume 5 Number 8 September 1987

BBC / ACORN

COMPUTER USER GROUP NZ (inc)

EDITORIAL

Just made it, another great issue with a wide range of contributions.

On behalf of all our readers, may I thank all the contributors this month -

F J Andrew	Wellington
Jane C	Tauranga
Peter Cheyne	Kaikoura (two)
Anne Flower	Wellington (two)
John Gleave	Christchurch
Don McDonald	Wellington
Steven Webb	Wellington
Terry Webb	Wellington
Local User Groups	Wellington
	Hawkes Bay

Can I draw your attention, particularly newer members, to the offer being publicized in the President's Report to supply back copies of *Beeblet* at the cost of post and packing. You couldn't get a better offer than that. Not all issues are available, and those that are will go to the first applicants.

And for those who missed the ads last month, they are back again. Prospective advertisers are to be encouraged for the benefit of both national and local users. As I have said before, we cannot make advertising rates much cheaper. Readers can be assured that advertisements do not displace contributions.

Happy Computing.

DISCLAIMER

The views of reviewers, contributors and/or Editorial committee are not necessarily shared by

THE BBC/ACORN COMPUTER USER GROUP NZ INC.

FROM YOUR PRESIDENT'S DESK

Nine months of the year gone already and this also means that this committee is half way through it's current term. I would like to take the opportunity to pass on to you what your committee has been doing for the last six months.

1. We have almost completely changed our administrative procedures and now use only commercial software to keep track of members' names, addresses, etc., and our finances. This was done to get away from the problem of trying to get to grips with an undocumented home made system each time the committee changed.

For your information we now use *Intersheet* for the finances and *Viewstore* as our members database. However, this should not be taken as a recommendation of these products.

2. The *Beeblet* magazine has been boosted to ensure it caters for members needs. We still suffer from a lack of contributions at times but we have still managed to put out a monthly issue. The release date has been changed to cater for Local User Groups who wish to advertise their meetings. Unfortunately, during August, our printer had a hic-up and we also had database problems sorting out a few people who had paid twice. Even with the best will in the world we sometimes slip up.

Advertising is accepted free of charge (although a donation is welcomed) to publicize products available within New Zealand.

3. The cost of joining the National User Group has been reduced from $30.00 to only $20.00. This, despite the increases in costs for producing and posting *Beeblet* to you. We reduced costs to answer criticism that $30.00 was too much to pay for an monthly magazine delivered to your door.

4. In the final analysis, the value of *Beeblet* lies more in its content than its cost. And for that we are reliant on contributions. I would like to personally thank those who have contributed to *Beeblet*; readers have benefitted from a wide range of topics in the last few months.

Several Local User Groups have been visited by committee members in an effort to generate interest in *Beeblet* and the National User Group. Offers have been made, and are still open, to publish information from special interest groups. While in Auckland, I was asked if *Beeblet* could include information from teachers relating to using the BBC in education. I answered yes, and the answer is still yes, but where is the information?

Any special interest group is welcome to use Beeblet as long as their activities include using the BBC computer.

5. Back issues of Beeblet are available free of charge to all members. If they have to be posted then a charge for the postage will have to be made. Over the years we have built up large stocks of Beeblet which the committee feel would be better being used by you than sitting in storage somewhere.

As you can see your National User Group committee has been working hard to improve the services offered by the National User Group. We have had some very good responses from Local User Groups who have thrown in their lot with the National Group. For a while now we have been concerned that Local User Groups go to all the effort and expense of producing a newsletter or magazine when the National Group will do so for no charge to the Local Group. This will release Local Group funds for a better use.

The points raised in my last report are still valid and I would like to re-emphasize the following:-

Unless BBC computer owners show a united front it will not be possible to convince dealers to import software or other products necessary to advance the use of or improve the facilities for the computer. Dealers need to have some idea as to the size of the market before they import products. Neither will it be possible to convince persons intending purchase of a BBC computer that this purchase is a good idea. User Group support is very important to first time buyers of computers. What is required is for all owners to get behind the National Group. Further more you should challenge us to provide your needs.

Yours for better computing,

Neville Hudson
President of the National User Group.

……

CROSS HATCH GENERATOR

By John Andrew, Wellington

The short program below will generate a cross hatch pattern on your screen. This pattern is used by Television Technicians when checking for convergence of the primary colours (red, green and blue). The same pattern can be used to check for linearity, purity and ghosting. I have used it to try and improve the performance of an old 26 inch Murphy colour set. An onslaught was made on the dying TV, with direct connections to output circuitry and tweaking of the convergence. I can now vaguely read 80 column text. The conversion process is not recommended but the program may be of use in determining the usefulness of old colour TVs for monitor service.

```
 10 REM Program to generate cross hatch on a colour
    screen to check colour convergence.
 20 REM SAVE "C.HATCH"
 30 REM F.J. Andrew 6 June 1987
 40 MODE0
 50 PRINT"COLOUR NUMBERS ARE"''
 60 PRINT"1 = RED"'"2 = GREEN"'"3 = GREEN and RED"'
    "4 = BLUE"
 70 PRINT"5 = RED and BLUE"'"6 = BLUE and GREEN"'
    "7 = WHITE"
 80 REPEAT : REM Allow user to change colours easily
 90 REPEAT:C=GET:UNTIL C>&30 AND C<&38 : REM Only allow
    numbers 1-7 (ASCII &31 to &37)
100 VDU19,1,C-&30,0,0,0 :REM Change logical colour
110 FOR I%=0 TO 1279 STEP 32 : REM step 32 pixels at a
    time
120 PLOT 4,I%,0 : REM Position cursor
130 DRAW I% ,1023 :REM Draw verticals
140 NEXT I%
150 PLOT4,1279,0: DRAW 1279,1023 : REM Add right
    boundary
160 FOR I%=0 TO 1023 STEP 32
170 PLOT 4,0,I% : REM Position cursor
180 DRAW 1279,I% : REM Draw horizontals
190 NEXT I%
200 PLOT4,0,1023:DRAW 1279,1023 : REM Add top boundary
210 UNTIL TRUE = FALSE : REM Impossible, so go back to
    REPEAT at line 80
220 END                    : REM Ploy to avoid GOTO !!
```

You too can win at LOTTO with a.....

LOTTO NUMBER GENERATOR

by Terry Webb, Wellington.

The inspiration for this program came after a visit to the Wellington Computer Show recently. At the show I saw a small device about the size of a pocket calculator, complete with LCD display, which generated six random numbers. Rather than invest in this expensive item I thought I would exploit the power of the Beeb.

I have tried to structure the program so that it is self-explanatory. The interesting feature is a check (lines 330 to 350) to ensure that the number has not already been generated. Lotto does not allow repeating numbers on the same board.

If you own an Electron change the Mode to 6, remove the FOR NEXT in PROCheading, and remove all Teletext control codes (CHR$129-159).

A percentage of any winnings resulting from use of this program will be gratefully accepted by the author.

```
 10 REM --WIN AT LOTTO--
 20 REM    by Terry W
 30 :
 40 MODE7
 50 DIMX(6)
 60 PROCheading
 70 VDU28,0,24,39,3
 80 @%=&00004
 90 PROCboards
100 PROCgenerate
110 GOTO 70
120 :
130 DEFPROCheading
140 FOR heading%=1 TO 2
150 PRINTCHR$131;CHR$157;CHR$141;CHR$129;TAB(11);
    "LOTTO NUMBER SELECTOR"
160 NEXT
170 ENDPROC
```

..... *LOTTO* >>>

```
 180 :
 190 DEFPROCboards
 200 CLS
 210 REPEAT
 220 PRINTTAB(2,0)CHR$134
     "Enter number of boards (4-10)      "
 230 INPUTTAB(33,0) boards%
 240 UNTIL boards%>=4 AND  boards%<=10
 250 ENDPROC
 260 :
 270 DEFPROCgenerate
 280 FOR loop%=1 TO boards%
 290 A=1
 300 REPEAT
 310 T=RND(39)+1
 320 check%=1
 330 IF X(check%)=T THEN 31Q
 340 check%=check%+1
 350 IF check%<6 THEN 330
 360 X(A)=T
 370 A=A+1
 380 UNTIL A>6
 390 PROCsort
 400 PROCdisplay
 410 NEXT loop%
 420 PRINT'CHR$130;
     "X to end, or any other key to rerun ";:G$=GET$
 430 IF G$="X" OR G$="x" THEN VDU26:CLS:END ELSE ENDPROC
 440 :
 450 DEFPROCsort
 460 FOR J=0 TO 5
 470 FOR K=J+1 TO 6
 480 IF X(J)<X(K) THEN 500
 490 R=X(J):X(J)=X(K):X(K)=R
 500 NEXT K
 510 NEXT J
 520 ENDPROC
 530 :
 540 DEFPROCdisplay
 550 PRINT'CHR$134"Board ";CHR$(64+loop%);CHR$135;"  ";
 560 FOR P=1 TO 6
 570 PRINT,X(P);
 580 NEXT P
 590 PRINT
 600 ENDPROC
```

....

BIGTEXT

by Peter Cheyne, Kaikoura

BIGTEXT is a PROCedure that can be used to produced text of any size on the screen using the standard font. It can be used with any of the graphics screen modes.

The PARAMETERS to be passed to the PROC are:

W$ - the string to be printed.

X% - the x coordinate of the "origin" of the text. (Normally this is the lower left hand corner of the text, but if the text is printed backwards or upside down it may be the top or right hand corner.)

Y% - the y coordinate of the "origin" of the text. (See note above).

DX% - the horizontal size of each character pixel. Negative values produce backwards (mirror image) writing.

DY% - the vertical size of each character pixel. Negative values produce up-side-down text.

The PROC uses a little routine that I don't understand (line 210 and part of 240) to determine the make-up of the character to be printed and then simply draws a rectangle with the dimensions DX% x DY% for each character element.

It is important to be careful positioning the text on the screen as the screen resolution can change the appearance of the character.

I am sure this must be easily converted into machine code which would make it much quicker. (Just think how much information could be crammed onto a screen using the "Microprint" option!)

```
  10 REM GIANT PRINT by PJC
  20 :
  30 MODE0
  40 PROCbigtext("BIG",250,700,30,30)
  50 PROCbigtext("TEXT",100,400,30,30)
  60 PROCbigtext("Reflections",100,300,4,4)
  70 PROCbigtext("Reflections",100,303,4,-4)
  80 PROCbigtext("BACKWARDS",1200,250,-8,8)
  90 PROCbigtext("UPSIDE DOWN & BACKWARDS",1200,203,-6,-4)
 100 PROCbigtext("Hello",0,80,30,6)
 110 PROCbigtext("MINIPRINT",480,28,2,2)
 120 PROCbigtext("MICROPRINT",500,12,1,1)
 130 END
 140 :
 150 DEF PROCbigtext(W$,X%,Y%,DX%,DY%)
 160 LOCAL H%,NX%,NY%,J%,L%,I%
 170 FOR H%=1 TO LEN W$
 180 NX%=X%+8*(H%-1)*DX%: NY%=Y%
 190 FOR J%=7 TO 0 STEP -1
 200 NX%=NX%+DX%: NY%=Y%
 210 IF ASC MID$(W$,H%,1)>223 L%=&C00+8*(ASC
     MID$(W$,H%,1)-224) ELSE L%=&BF00+8*ASC MID$(W$,H%,1)
 220 FOR I%=7 TO 0 STEP -1
 230 NY%=NY%+DY%
 240 IF (L%?I%AND2^J%) MOVE NX%,NY%: MOVE
     NX%+DX%-SGN(DX%),NY%: PLOT 85,NX%,NY%+DY%-SGN(DY%):
     PLOT 85,NX%+DX%-SGN(DX%),NY%+DY%-SGN(DY%)
 250 NEXT:NEXT:NEXT
 260 ENDPROC
```

~~~~~~

For Sale:

VIEWSPELL, Version 1.0: ROM + 80 track disc	$50
VIEWPLOT, 40/80 track disc, DFS	$45
VIEW 3.0, ROM	$50
VIEW INDEX for BBC, 40/80 track	$20
All complete with manuals.	
VIEWSHEET manual only	$15

Telephone John (03) 321-278 evenings.

~~~~~~

SAVING MASTER RED KEYS

Dear Gavin Flower,

> *I don't like this new Master - I can't use my *LOAD KEYS anymore. The blasting computer doesn't have its RAM where it ought to. Please write me a simple little program to have the same effect as the *SAVE KEYS B00 +100 that I'm used to. I can't get Wordwise definitions right using your *BUILD method for key definitions.*

Love, Anne.

This reduced Gavin to machine code for the first time on an Acorn machine!

HOW TO USE IT:
Type in the listings below, and save each under some innocuous name that you will recognise before you go any further. Then RUN. The listing KLOAD automatically saves the object code produced onto disc, so have one ready. So does KSAVE.

From now on to save a set of key definitions, type in "*KSAVE file" and the computer will hunt through its ROMS then a disc for this command. LOADing a previously-saved set of definition is similar:
*KLOAD WOKEYS (or whatever relevant filename).

The programs here are fully REMmed for those who want to understand how it shifts data in and out of different parts of memory. They have been fully tested using DFS. (Not ADFS!)

```
 10 REM  B.KSAVE
 20 REM  Program to compile "KSAVE filename"
 30 REM  Saves MASTER softkey definitions to named file
 40 REM   - reloaded by "KLOAD filename"
 50 DIM code_area 511
 60 tuwa = &DD00         :REM Transient Utility Work Area
 70 buffer = (tuwa AND&FF00) +&100  :REM temporary
    buffer
 80 romsel = &FE30       :REM ROM select register
 90 romsel_copy = &F4    :REM os copy of romsel
100 mos_ram = &8000      :REM where soft keys are stored
110 osfind = &FFCE
```

.... *B.KSAVE* Contd. >>

```
120 osgbpb = &FFD1
130 from = &DA             :REM "ZTEMPA" - MASTER ref. page
                                E.4-4
140 to = &DC               :REM "ZTEMPB"
150 *REMOVE KSAVE
160 *REMOVE KEYS
170 PROC_assemble
180 *KSAVE KEYS
190 VDU14
200 *DUMP KEYS
210 VDU15
220 END
230
240 DEF PROC_assemble
250 LOCAL I
260 FOR I = 4 TO 7 STEP 3
270   O% = code_area
280   P% = tuwa
290   [
300   OPT I
310   \
320   \ set X,Y to point to file name
330   \
340                    TYA
350                    CLC
360                    ADC &F2
370                    TAX
380                    LDY &F3
390   \
400   \ open file for output
410   \
420                    LDA #&80
430                    JSR osfind
440                    STA block
450                    CMP #0
460                    BNE set_pointers
470                    BRK
480                    EQUB &FF
490                    EQUS "open failure"
500                    EQUB 0
510   \
520   \ set from/to pointers
530   \
540   .set_pointers
550                    LDA #mos_ram MOD256
560                    STA from
```

.... *B.KSAVE* Contd. >>>

.... *B.KSAVE* Contd. >>>

```
 570                     LDA #mos_ram DIV 256
 580                     STA from+1
 590                     LDA #buffer MOD 256
 600                     STA to
 610                     LDA #buffer DIV 256
 620                     STA to+1
 630    \
 640    \ Page in softkey area
 650    \
 660    .get_block
 670                     LDA romsel_copy
 675    \ MASTER ref. page E.4-9
 680                     PHA
 690                     ORA #&80
 700                     STA romsel_copy
 710                     STA romsel
 720    \
 730    \ move block from MOS RAM to buffer
 740    \
 750                     LDY #0
 760    .next_byte
 770                     LDA (from),Y
 780                     STA (to),Y
 790                     DEY
 800                     BNE next_byte
 810    \
 820    \ restore ROMSEL
 830    \
 840                     PLA
 850                     STA romsel_copy
 860                     STA romsel
 870    \
 880    \ reset data address & number of bytes in "block"
 890    \
 900                     LDA #buffer AND&FF
 910                     STA block+1
 920                     LDA #buffer DIV256
 930                     STA block+2
 940                     LDA #1
 950                     STA block+6
 960    \
 970    \ write buffer
 980    \
 990                     LDA #02
1000                     LDX #block AND&FF
```

.... *B.KSAVE* Contd. >>>

.... *B.KSAVE* Contd. >>>

```
 1010                    LDY #block DIV256
 1020                    JSR osgbpb
 1030   \
 1040   \ check for sucessful I/O
 1050   \
 1060                    BCC ok
 1070                    BRK
 1080                    EQUB &FF
 1090                    EQUS "I/O write failure"
 1100                    EQUB 0
 1110   \
 1120   \ increment 'from' page pointer
 1130   \
 1140   .ok
 1150                    INC from+1
 1160   \
 1170   \ exit if finished - need to write 4*256 bytes
 1180   \
 1190                    DEC count
 1200                    BNE get_block
 1210   \
 1220   \ close file
 1230   \
 1240                    LDA #0
 1250                    LDY block
 1260                    JSR osfind
 1270   \
 1280   \ return
 1290   \
 1300                    RTS
 1310   \
 1320   .block           EQUB 0          \ channel
 1330                    EQUD buffer     \ data address
 1340                    EQUD 256        \ number of bytes
 1350                    EQUD 0          \ file pointer
 1360   \
 1370   .count           EQUB 4
 1380   \
 1390   ]
 1400 NEXT
 1410 OSCLI("SAVE KSAVE "+STR$~code_area +" "+ STR$~O% +"
      "+ STR$~tuwa +" "+ STR$~tuwa)
 1420 ENDPROC
```

....End of *B.KSAVE* ....

..... <<<<<   >>>>> .....

```
 10 REM  B.KLOAD
 20 REM  Program to compile "KLOAD filename"
 30 REM  (Loads in softkey definitions saved by KSAVE)
 40 DIM code_area 511
 50 tuwa = &DD00          :REM Transient Utility Work Area
 60 buffer = (tuwa AND&FF00) +&100  :REM temporary
    buffer
 70 romsel = &FE30        :REM ROM select register
 80 romsel_copy = &F4     :REM os copy of romsel
 90 mos_ram = &8000       :REM where soft keys are stored
100 osfind = &FFCE
110 osgbpb = &FFD1
120 to = &DA
130 from = &DC
140 *REMOVE KLOAD
150 PROC_assemble
160 *KLOAD KEYS
170 END
180
190 DEF PROC_assemble
200 LOCAL I
210 FOR I = 4 TO 7 STEP 3
220   O% = code_area
230   P% = tuwa
240   [
250   OPT I
260   \
270   \ set X,Y to point to file name
280   \
290                TYA
300                CLC
310                ADC &F2
320                TAX
330                LDY &F3
340   \
350   \ open file for input
360   \
370                LDA #&40
380                JSR osfind
390                STA block
400                CMP #0
410                BNE set_pointers
420                BRK
430                EQUB &FF
440                EQUS "open failure"
450                EQUB 0
460   \
```

... *B.KLOAD* Contd. >>>

... *B.KLOAD* Contd. >>>

```
470   \ set to/from pointers
480   \
490   .set_pointers
500                    LDA #mos_ram MOD256
510                    STA to
520                    LDA #mos_ram DIV 256
530                    STA to+1
540                    LDA #buffer MOD 256
550                    STA from
560                    LDA #buffer DIV 256
570                    STA from+1
580   \
590   \ read buffer
600   \
610   .next_block
620                    LDA #04
630                    LDX #block AND&FF
640                    LDY #block DIV256
650                    JSR osgbpb
660   \
670   \ check for sucessful I/O
680   \
690                    BCC ok
700                    BRK
710                    EQUB &FF
720                    EQUS "I/O read failure"
730                    EQUB 0
740   \
750   \ Page in softkey area
760   \
770   .ok              LDA romsel_copy
780                    PHA
790                    ORA #&80
800                    STA romsel_copy
810                    STA romsel
820   \
830   \ move block to MOS RAM from buffer
840   \
850                    LDY #0
860   .next_byte
870                    LDA (from),Y
880                    STA (to),Y
890                    DEY
900                    BNE next_byte
910   \
920   \ restore ROMSEL
```

... *B.KLOAD* Contd. >>>

... *B.KLOAD* Contd. >>>

```
 930   \
 940                   PLA
 950                   STA romsel_copy
 960                   STA romsel
 970   \
 980   \ increment to pointer
 990   \
1000                   INC to+1
1010   \
1020   \ reset data address & number of bytes in "block"
1030   \
1040                   LDA #buffer AND&FF
1050                   STA block+1
1060                   LDA #buffer DIV256
1070                   STA block+2
1080                   LDA #1
1090                   STA block+6
1100   \
1110   \ exit if finished - need to read 4*256 bytes
1120   \
1130                   DEC count
1140                   BNE next_block
1150   \
1160   \ close file
1170   \
1180                   LDA #0
1190                   LDY block
1200                   JSR osfind
1210   \
1220   \ return
1230   \
1240                   RTS
1250   \
1260   .block          EQUB 0          \ channel
1270                   EQUD buffer     \ data address
1280                   EQUD 256        \ number of bytes
1290                   EQUD 0          \ file pointer
1300   \
1310   .count          EQUB 4
1320   \
1330   ]
1340 NEXT
1350 OSCLI("SAVE KLOAD "+STR$~code_area +" "+
     STR$~O% +" "+ STR$~tuwa+" "+ STR$~tuwa)
1360 ENDPROC
```

~~~~~~
~~~~~~

DOUBLE HEIGHT DEMONSTRATION

by Jane C., Tauranga

This program demonstrates enlarged lettering in the various modes. The program is largely (to coin a phrase) self explanatory, so without further ado.....

```
 10 REM DOUB/HT DEM CH."BIGPROG"
 20 MODE7
 30 mo%=7
 40 PROCbig7(5,1,CHR$130,"DOUBLE HEIGHT DEMONSTRATION")
 50 PROCbig7(0,5,CHR$134,"This is MODE 7 using PROCbig7")
 60 PROCbig7(0,8,CHR$131,"I will follow it with questions
    which")
 70 PROCbig7(0,10,CHR$131,"you will answer using
    PROCbiginput")
 80 PROCbig7(0,13,CHR$130,"What is your name")
 90 col$=CHR$135
100 PROCbiginput(19,13)
110 N$=I$
120 PROCbig7(0,16,CHR$133,"Hello "+N$)
130 PROCbig7(1,22,CHR$135,"PRESS ANY KEY FOR THE OTHER
    MODES")
140 G$=GET$
150 MODE1
160 mo%=0
170 COLOUR1
180 PROCbig(0,6,"The other modes use PROCbig but the")
190 PROCbig(0,8,"same PROCbiginput")
200 COLOUR2
210 PROCbig(0,12,"Of course they do not all have the")
220 PROCbig(0,14,"full range of colours")
230 COLOUR3
240 PROCbig(0,18,"This is Modes 1 or 4")
250 COLOUR1
260 PROCbig(0,22,"Enter your name")
270 COLOUR2
280 PROCbiginput(17,22)
290 COLOUR3
300 PROCbig(12,28,"PRESS ANY KEY")
310 G$=GET$
320 MODE2
330 mo%=0
340 COLOUR1
```

.... *BIGPROG* >>>

.... *BIGPROG* >>>

```
350 PROCbig(0,8,"This")
360 COLOUR2
370 PROCbig(5,8,"is")
380 COLOUR3
390 PROCbig(0,11,"Modes")
400 COLOUR4
410 PROCbig(6,11,"2")
420 COLOUR5
430 PROCbig(8,11,"or")
440 COLOUR6
450 PROCbig(11,11,"5")
460 COLOUR13
470 PROCbig(0,15,"Enter your name")
480 COLOUR7
490 PROCbiginput(0,18)
500 COLOUR1
510 PROCbig(3,25,"PRESS ANY KEY")
520 G$=GET$
530 MODE6
540 mo%=0
550 PROCbig(0,8,"This is Mode 6 & probably not much use")
560 PROCbig(0,13,"Enter your name")
570 PROCbiginput(17,13)
580 PROCbig(0,18,"Do you want to see the program again?")
590 G$=GET$
600 IF G$="Y"ORG$="y" GOTO 20
610 CLS
620 END
630
640 DEFPROCbig7(X%,Y%,C$,M$)
650 FORL%=Y%TOY%+1:PRINTTAB(X%,L%)CHR$141;C$;M$:NEXT
660 ENDPROC
670
680 DEFPROCbig(H%,V%,M$)
690 LOCAL N
700 A%=&A:X%=0:Y%=&A
710 D=&A00
720 FOR N=1 TO LEN(M$)
730 B$=MID$(M$,N,1)
740 ?D=ASC(B$)
750 CALL (&FFF1)
760 VDU23,240,D?1,D?1,D?2,D?2,D?3,D?3,D?4,D?4
770 VDU23,241,D?5,D?5,D?6,D?6,D?7,D?7,D?8,D?8
780 PRINTTAB(H%+N,V%)CHR$240;TAB(H%+N,V%+1)CHR$241
790 NEXT
800 ENDPROC
```

....*BIGPROG* >>>

.... BIGPROG >>>

```
 810
 820 DEFPROCbiginput(XX%,YY%)
 830 IF mo%=7 W%=XX%+2 ELSE W%=XX%+1
 840 I$=""
 850 REPEAT
 860 VDU31,W%,YY%
 870 G%=GET
 880 IF I$=""ANDG%=13 GOTO 870
 890 IF G%=127 AND I$<>"" I$=LEFT$(I$,len%-1):PROCerase:
     W%=W%-1:VDU31,W%,YY%:GOTO920 ELSE IF G%=127 GOTO870
 900 G$=CHR$(G%)
 910 IF G%<>13 I$=I$+G$
 920 IF G%<>13 AND mo%=0 PROCbig(XX%,YY%,I$)
 930 IF G%<>13 AND mo%=7 PROCbig7(XX%,YY%,col$,I$)
 940 len%=LEN(I$)
 950 IF G%<>127 W%=W%+1
 960 UNTILG%=13
 970 ENDPROC
 980
 990 DEFPROCerase
1000 IF mo%=0 PROCbig(XX%,YY%,STRING$(len%," "))
1010 IF mo%=7 PROCbig7(XX%,YY%,CHR$128,STRING$(len%," "))
1020 ENDPROC
```

An interesting one liner......

PATTERN

by Peter Cheyne, Kaikoura.

PATTERN is a totally trivial programme that creates some quite startling patterns simply by drawing a square (perhaps modified in some way) and then rotates it. The original square may be modified by removing or changing the direction on any of the sides.

Two questions for the more mathematically minded: How many different patterns are possible? And, how long could one spend gazing at the screen?

```
 10 MODE0:X1=0:Y1=0:X2=0:Y2=1023:REPEAT:MOVEX1,Y1:
    DRAWX2,Y2:DRAW1023-X1,1023-Y1:DRAW1023-X2,1023-Y2:
    DRAWX1,Y1:D1=(X2-X1)/60:D2=(Y2-Y1)/60:X1=X1+D1:
    Y1=Y1+D2:X2=X2+D2:Y2=Y2-D1:UNTIL(X2-X1)^2+(Y2-Y1)^2<1
```

ZOOM Pixel Editor

by Steven Webb, Wellington.

This program allows you to "touch-up" a graphics screen pixel-by-pixel so as to enhance/modify selected small areas. It is a useful addition for those of you who use a graphics design program such as Colour Icon (*A&B Computing*) or Freehand (*Acorn User*). The program is rather long, but I hope this doesn't deter anyone, as it is extremely useful.

Basically, the program allows the user to select a small area of a screen (previously *SAVEed) to "enlarge" and edit, after which it is reduced back to normal size and restored into the main picture.

The machine code at the end is a short routine for copying the top half of the screen to sideways RAM. This also restores the picture after editing. The program will not run unless sideways RAM is installed.

In its present state it is for use with the Solidisk 2MEG 128K expansion board, which they say is compatable with the Master 128. If you own a Solidisk old-style sideways RAM board, the following changes will enable it to work: delete lines 1970,1980,2190,2270,2480, and replace lines 2230-2260 with:

```
2230 LDA #15
2240 STA &FE62
2250 LDA #bank
2260 STA &FE60
```

It is designed to work with BASIC 1. The keys to use are displayed upon running the program. When the screen is saved, it is always given the filename P.PICTURE, so it must be renamed afterwards to prevent overwriting next time.

```
10 REM *** ZOOM Pixel Editor ***
20 REM
30 REM by Steven Webb
40 :
50 MODE7
```

.... *ZOOM* Pixel Editor >>>

```
 60 VDU23;8202;0;0;0;
 70 DIM logcol%(7),osc% 30
 80 PROCstart
 90 MODE mode
100 VDU23;8202;0;0;0;
110 FORA%=0TOmode
120 READ himem,pixels,ppb,step
130 NEXT A%
140 PROCassemble
150 DATA &30,640,2,&F8
160 DATA &30,320,4,&F8
170 DATA &30,160,8,&F8
180 DATA 0,0,0,0
190 DATA &58,320,4,&80
200 DATA &58,160,8,&80
210 IF yn$="Y" OR yn$="" THEN PROCoscli("LOAD
    "+filename$+" "+STR$~himem+"00")
220 *FX4,1
230 *OPT1,0
240 X%=0:Y%=0
250 PROCbox
260 REPEAT
270 IF INKEY-58 AND Y%<944 THEN PROCbox:
    Y%=Y%+(ppb*2/(INKEY-1+2)):PROCbox
280 IF INKEY-42 AND Y%>0 THEN
    PROCbox:Y%=Y%-(ppb*2/(INKEY-1+2)):
    PROCbox
290 IF INKEY-26 AND X%>0 THEN PROCbox:
    X%=X%-(ppb*2/(INKEY-1+2)):PROCbox
300 IF INKEY-122 AND X%<(1280-(20*ppb)) THEN PROCbox:
    X%=X%+(ppb*2/(INKEY-1+2)):PROCbox
310 inp%=INKEY(0)
320 IF inp%>47 AND inp%<56 THEN PROClogcol(inp%-48)
330 IF INKEY-74 THEN PROCzoom
340 IF INKEY-82 THEN PROCbox:
    PROCoscli("SAVE P.PICTURE "+STR$~himem+"00 7FFF"):
    PROCbox
350 UNTIL FALSE
360 :
370 DEFPROCbox
380 GCOL3,7
390 MOVEX%,Y%
400 DRAWX%+(20*ppb)-ppb,Y%
410 DRAWX%+(20*ppb)-ppb,Y%+80-ppb
420 DRAWX%,Y%+80-ppb
```

.... *ZOOM* Pixel Editor >>>

.... *ZOOM* Pixel Editor >>>

```
430 DRAWX%,Y%
440 ENDPROC
450 :
460 DEFPROClogcol(col%)
470 A%=logcol%(col%)
480 A%=A%+1:IF A%=8 THEN A%=0
490 logcol%(col%)=A%
500 VDU19,col%,logcol%(col%),0,0,0
510 FOR A%=0 TO 350:NEXT
520 ENDPROC
530 :
540 DEFPROCzoom
550 PROCbox
560 FORV%=19TO0STEP-1
570 FORH%=0TO19
580 ?(&900+(V%*20)+H%)=POINT(X%+(H%*ppb),Y%+(V%*4))
590 NEXT H%
600 NEXT V%
610 CALL copy
620 GCOL0,0
630 MOVE0,1023
640 MOVE480,1023:PLOT85,480,543
650 MOVE0,1023
660 PLOT85,0,543
670 GCOL0,7
680 FORV%=0TO20
690 MOVE V%*24,1023:DRAW V%*24,544
700 NEXT
710 FORH%=0TO20
720 MOVE 0,1023-(H%*24):DRAW 480,1023-(H%*24)
730 NEXT
740 FORV%=0TO19
750 FORH%=0TO19
760 col%=?(&900+((19-V%)*20)+H%)
770 IF col%=0 THEN GOTO840
780 GCOL0,col%
790 MOVEH%*24+ppb,1023-(V%*24)-4
800 MOVE(H%+1)*24-ppb,1023-(V%*24)-4
810 PLOT85,(H%+1)*24-ppb,1023-((V%+1)*24)+4
820 MOVEH%*24+ppb,1023-(V%*24)-4
830 PLOT85,H%*24+ppb,1023-((V%+1)*24)+4
840 NEXT:NEXT
850 x%=0:y%=0
860 PROCbox2
870 REPEAT
```

.... *ZOOM* Pixel Editor >>>

```
 880 IF INKEY-58 AND y%>0 THEN PROCbox2:y%=y%-1:PROCbox2
 890 IF INKEY-42 AND y%<19 THEN PROCbox2:y%=y%+1:
     PROCbox2
 900 IF INKEY-26 AND x%>0 THEN PROCbox2:x%=x%-1:PROCbox2
 910 IF INKEY-122 AND x%<19 THEN PROCbox2:x%=x%+1:
     PROCbox2
 920 IF INKEY-40 THEN PROCcolour(0)
 930 IF INKEY-49 THEN PROCcolour(1)
 940 IF INKEY-50 THEN PROCcolour(2)
 950 IF INKEY-18 THEN PROCcolour(3)
 960 IF INKEY-19 THEN PROCcolour(4)
 970 IF INKEY-20 THEN PROCcolour(5)
 980 IF INKEY-53 THEN PROCcolour(6)
 990 IF INKEY-37 THEN PROCcolour(7)
1000 UNTIL INKEY-74
1010 CALL uncopy
1020 FORV%=19TO0STEP-1
1030 FORH%=0TO19
1040 GCOL0,?(&900+(V%*20)+H%)
1050 PLOT69,X%+(H%*ppb),Y%+(V%*4)
1060 NEXT H%
1070 NEXT V%
1080 PROCbox
1090 *FX15
1100 ENDPROC
1110 :
1120 DEFPROCbox2
1130 GCOL3,3
1140 MOVEx%*24,1023-y%*24
1150 DRAWx%*24+24,1023-y%*24
1160 DRAWx%*24+24,1023-y%*24-24
1170 DRAWx%*24,1023-y%*24-24
1180 DRAWx%*24,1023-y%*24
1190 ENDPROC
1200 :
1210 DEFPROCcolour(col%)
1220 ?(&900+((19-y%)*20)+x%)=col%
1230 GCOL0,col%
1240 MOVEx%*24+ppb,1023-(y%*24)-4
1250 MOVE(x%+1)*24-ppb,1023-(y%*24)-4
1260 PLOT85,(x%+1)*24-ppb,1023-((y%+1)*24)+4
1270 MOVEx%*24+ppb,1023-(y%*24)-4
1280 PLOT85,x%*24+ppb,1023-((y%+1)*24)+4
1290 ENDPROC
1300 :
```

.... *ZOOM* Pixel Editor >>>

```
1310 DEFPROCoscli($osc%)
1320 oldX=X%:oldY=Y%
1330 X%=osc% MOD 256
1340 Y%=osc% DIV 256
1350 CALL&FFF7
1360 X%=oldX:Y%=oldY
1370 ENDPROC
1380 :
1390 DEFPROCstart
1400 FORA%=0TO1
1410 PRINTCHR$133;CHR$157;CHR$141;CHR$(130+A%);
     TAB(11);"ZOOM Pixel Editor"
1420 NEXT
1430 PRINT'TAB(13);"by Steven Webb"
1440 FOR A%=5 TO 23
1450 PRINTTAB(0,A%)CHR$134
1460 NEXT
1470 VDU28,1,23,39,5:CLS
1480 PRINT TAB(0,1)"Which SWR bank (15) ?        ";
1490 INPUT TAB(22,1) bank$
1500 IF bank$="" THEN bank$="15":PRINTTAB(22,1)bank$
1510 bank=VAL(bank$)
1520 IF bank>15 OR bank<0 THEN GOTO 1480
1530 PRINT TAB(0,3)"Which screen mode (1) ?    ";
1540 INPUTTAB(25,3) mode$
1550 IF mode$="" THEN mode$="1":PRINTTAB(25,3)mode$
1560 mode=VAL(mode$)
1570 IF mode<0 OR mode=3 OR mode>5 THEN GOTO 1530
1580 PRINTTAB(0,5)"Load a screen (Y) ?    "
1590 INPUTTAB(21,5) yn$
1600 IF yn$<>"Y" AND yn$<>"" THEN GOTO1710
1610 CLS
1620 INPUT"Load from drive? (0) "dr$
1630 IF dr$="" THEN PRINTTAB(21,0)"0":dr$="0"
1640 PROCoscli("DRI."+dr$)
1650 INPUT'"Would you like a catalogue? (N) "ans$
1660 IF ans$="Y" THEN CLS:VDU14:*.
1670 PRINT''"Enter screen filename :";CHR$131;
1680 INPUTTAB(24) filename$
1690 IF LENfilename$>7 AND MID$(filename$,2,1)<>"." THEN
     PRINTCHR$129"Filename too long!!":GOTO1670
1700 IF MID$(filename$,2,1)="." AND LENfilename$>9 THEN
     PRINTCHR$129"Filename too long!!":GOTO1670
1710 CLS
1720 PRINTCHR$130;TAB(18)"KEYS:"
```

.... *ZOOM* Pixel Editor >>>

```
1730 PRINT'CHR$133TAB(10)"In Full-screen mode:"
1740 PRINTTAB(9)"cursor keys...move box"
1750 PRINTTAB(9)"   +<SHIFT>...faster"
1760 PRINTTAB(9)"<RET>..enter ZOOM mode"
1770 PRINTTAB(9)"0-7.....change logical"
1780 PRINTTAB(9)"               colours"
1790 PRINTTAB(9)"S..........save screen"
1800 PRINT'CHR$133TAB(14)"In ZOOM mode:"
1810 PRINTTAB(9)"0-7...change pixel col"
1820 PRINTTAB(9)"<RET>......enter Full-"
1830 PRINTTAB(9)"           Screen mode"
1840 PRINT''CHR$131;TAB(12)"Any key to begin"
1850 G=GET
1860 ENDPROC
1870 :
1880 DEFPROCassemble
1890 FORPASS=0TO2STEP2
1900 P%=&A96
1910 [OPT PASS
1920 .uncopy
1930 LDA &F4
1940 STA &82
1950 LDA #bank
1960 STA &FE30
1970 STA &FE32
1980 STA &F4
1990 LDA #step:STA &87
2000 LDA #0:STA &85
2010 LDA #&80:STA &86
2020 LDA #0:STA &83
2030 LDA #himem:STA &84
2040 LDX #&27
2050 LDY #0
2060 .loop
2070 LDA (&85),Y
2080 STA (&83),Y
2090 INY
2100 CPY &87
2110 BNE loop
2120 INC &84
2130 INC &86
2140 DEX
2150 BNE loop
2160 .finish
2170 LDA &82
2180 STA &FE30
2190 STA &F4
2200 RTS
2210 :
2220 .copy
2230 LDA &F4
2240 STA &82
2250 LDA #bank
2260 STA &FE30
2270 STA &FE32
2280 STA &F4
2290 LDA #step:STA &87
2300 LDA #0:STA &85
2310 LDA #&80:STA &86
2320 LDA #0:STA &83
2330 LDA #himem:STA &84
2340 LDX #&27
2350 LDY #0
2360 .loop2
2370 LDA (&83),Y
2380 STA (&85),Y
2390 INY
2400 CPY &87
2410 BNE loop2
2420 INC &84
2430 INC &86
2440 DEX
2450 BNE loop2
2460 LDA &82
2470 STA &FE30
2480 STA &F4
2490 RTS
2500 ]
2510 NEXT
2520 ENDPROC
```

AVAILABLE FROM :-

MICRO INTERFACE (NZ) LTD

P.O. BOX 14441
KILBIRNIE
WELLINGTON.
a/h 882-444   STARNET 6009:UGR008

Please Note: All prices include GST

INTERWORD	- 32k Word-Processor for only $175.82 + $3.00p&p.
INTERSHEET	- ROM based spreadsheet for only $175.82 + $3.00p&p.
INTERCHART	- ROM based chart, graph and plotting program for only $118.20 + $3.00 p&p.
INTERBASE	- ROM based database for only $220.00 + $3.00 p&p.
SPELLMASTER	- 128k ROM based spelling checker compatible with Wordwise, Interword and View. Only $186.85 $3.00 p&p.
MASTERFILE II	- Random access database for the ADFS. Both 5 1/4" and 3 1/2" disc versions only $88 + $3.00 p&p.
MINI OFFICE II	- 5 1/4" disc based package for the BBC B and B+. Contains a word processor, database, spreadsheet, graphics program, communications program and a label printing program. Usually $63.70 but kept on special for only $53.70 until the end of September, + $3.00 p&p.
BOX OF TREASURES	- Disc based educational adventure package. 5 1/4" disc $87.09 or 3 1/2" disc $90.00 + $3.00 p&p.
FLOWERS OF CRYSTAL	- Disc based educational adventure package. 5 1/4" disc $68.48 or 3 1/2" disc $72.00 + $3.00 p&p.

GRANNY'S GARDEN - Disc based educational adventure package. 5 1/4" disc $48.84 or 3 1/2" disc $52.00 + $2.00 p&p.

SPACEX - Another disc based educational adventure package. 5 1/4" disc $43.26 or 3 1/2" disc $45.00 + $2.00 p&p.

THE MAGIC SWORD - Disc based educational adventure package for the BBC B. 80 track 5 1/4" disc $47.51 + $2.00 p&p.

ELITE! - Disc based games package, enhanced version, for the BBC and Master. Now only $53.29 +$3.00 p&p. Electron cassette version $46.49 + $3.00 p&p.

REPTON 3, STRYKERS RUN, THRUST,
SUPERIOR COLLECTION VOLUME ONE,
SUPERIOR COLLECTION VOLUME TWO,
AROUND THE WORLD IN 40 SCREENS,
CITADEL, GALAFORCE, and RAVENSKULL.

- All games are 5 1/4" 40/80 track disc based games for the BBC and BBC Master. Now only $43.09 + $2.00 p&p each.
Some 3 1/2" disc versions available.

SOLDERLESS SIDEWAYS ROM SOCKET BOARD
- Expands the total possible number of ROMs in a BBC from 4 to 16. Only $147.50 + $3.00 p&p.

HCR MASTER ROM BOX - Expands Masters ROM sockets to twelve. Make better use of the Masters power. $325.00 + $5.00 p&p.

3 1/2" DISCS - 135 TPI only $75.00 + $3.00 p&p.

BEEBUG MAGAZINE - Flown in from England especially for you. Only $6.95 + 0.60 cents p&p.

Many other products available too numerous to list here.

WAVES CRASHING

Anne Flower, Wellington

Experimenting with the amplitude parameters of the Envelope statement, I inadvertently started using channel 0 (the noise channel) instead of one of the tune channels for testing different pitches. So I was surprised when I got not notes, but static-like sounds with certain pitches. The difference between the two envelopes used is that the first does not have any decay, whereas the second has. The two sets of SOUND statements are identical - the alteration of one parameter only in the Envelope is responsible for the difference.

```
10 REM shows the effects of the ALD parameter in an
   ENVelope
20 PRINT"Waterless waves crashing"
30 ENVELOPE1,6,0,0,0,1,0,0,30,-4,0,-4,126,0
40 SOUND0,1,20,50:SOUND0,1,30,60:SOUND0,1,30,40:
   SOUND0,1,20,80
50 wait=INKEY(1000)
60 PRINT"Watery waves"
70 ENVELOPE1,6,0,0,0,1,0,0,30,-4,0,-4,126,100
80 SOUND0,1,20,50:SOUND0,1,30,60:SOUND0,1,30,40:
   SOUND0,1,20,80
90 END
```

For Sale:

BBC Model B Computer.
65C02 3mHz Co-processor.
Twin Disc Drive: 80 track, Double sided with power supply.
High resolution NEC PC-8043B 12" Colour monitor

Price: $1700 ono.

Phone Dave Nuttall, (09) 591-509 (Auckland)

MASTERFILE II

A review

by John Gleave, Christchurch.

I have tried several databases including ViewStore which was reviewed in the August issue of *Beeblet* and have found that Masterfile II (a BEEBUG product) the easiest to use and sufficiently versatile for most purposes. Certainly ViewStore is more versatile but after having it on trial for a weekend I decided that it was just too complicated to master readily and that I would continue with Masterfile II although it is a little limited at times as each record is limited to one screen of text in mode 7.

The latest version is called ADFS Masterfile II and it gives some additional features not available in the DFS version - these are that form labels, tag labels and forms can be retained and labelled for reuse. This version makes use of the shadow ram for tag sorting allowing larger files to be tagged sorted - the length depends upon the amount of shadow ram in use.

Main Menu.

The main menu has options A to P which relate to setting up new files, searching records, sorting records, setting up printer configuration, designing forms, looking at and altering records - all very easy to use especially on a colour screen.

The first step is to select option A and enter the name of the database to be searched, sorted, altered or printed. If a name is selected which is not on the disc then the screen advises you to select option B to create a totally new file followed by option F to reserve sufficient space on the disc for the file containing the number of records and length and number of fields created under option B.

New Databases.

A new database is set up under option B where the length and title of each field is established along with the type of field. There is provision for strings, numbers (the number of decimal points may be specified), integers (no decimal points), and dates. This is to facilitate sorting.

Look at and Alter Data.

Under this option the record appears on the screen with its number in the top right and summarised instructions below. The field numbers and titles appear on the left of the screen with the data entered on the right. The length of each field is indicated by an < on the screen. Movement through the fields and records is by use of the cursor keys with provision to enter all fields on a record or only a selected one to alter data. A particular record can be moved to by typing Rn where n is the number of the record. Data is entered as strings and numbers can be recalculated individually or globally by moving to option L on the main menu.

Search.

One of the most important features of a database is to be able to search for selected records meeting criteria stipulated by the user. Masterfile permits search under a variety of criteria using one of more fields. The example given is a list of people with, inter alia, sex and height shown in separate fields. The computer can be asked to find the records of all females and this could be further reduced to all females over a certain height. Once a search has been completed only those records meeting the criteria will be printed out until the search parameters are cancelled.

Sorting.

Masterfile has two methods of sorting records. Under the full sort option the records are changed in order on the file. This takes time (as much as an hour for each fifty records sorted on more than one field) and the original file is lost so if it is required again a backup must be made before sorting. The tag sort is very much quicker but sorting can only be done on two fields at once. Under Masterfile II the tag sort is lost once the computer is turned off but under the ADFS version it can be retained and the required tag sort required obtained through the Management option. Sorts can be in either ascending or descending order.

Printout.

Individual records can be dumped direct from the look at or alter a record option.

The line length, left margin and printer controls can be set under option D before making a printout.

Using the print option from the main menu all the records can be printed out either horizontally across the page or vertically under one another. By making up a form first and if required using a tag sort all the fields or selected ones can be printed. There are options to print labels or forms. The forms are designed by the user from another option on the main menu - fields, dates, numbers, headings, calculations and text can be placed anywhere on the printout. One of the form commands enables data to be inserted into a previously prepared file (spooled from a word processor) containing a letter.

There is also provision to print labels.

Conclusions.

A simple to use versatile database which once in use requires little reference to the manual as many instructions appear on the screen. Apart from the limitations on the size of a record mentioned earlier my only criticism is that at the time I bought my copy there is no separate manual for the ADFS version and it was necessary to amend the manual from photo-copied supplements.

~~~~~~

HandiBags v Discs

Advice for Contributors using Discs.

One size of Handibag is just right for packing discs. Unfortunately they don't really provide enough protection against bending - when heavy parcels "lean" on them. One disc received recently was in a sad state.

It is recommended that the disc be taped to at least one piece of stiff card about 150mm square before being placed in the Handibag.

Special tip: - don't throw away the card used when your disc is returned to you - regard it as belonging to *Beeblet* and keep it for when you send in your next contribution.
~~~~~~

LOCAL USER GROUP NEWS

WELLINGTON

The August meeting, held on the 12th, was very open plan. People came and discussed any problems they were having or new software they had written or acquired. The biggest topic under discussion was what was wrong with the computer on the desk in the front. The best laid plans of mice and computer buffs are still prone to go wrong.

If you have anything that you would like to display to the Wellington User Group please feel free to do so. Contact phone numbers are on the back of this issue of Beeblet.

As usual, the User Group meetings are held in the Lecture Theatre, 1st floor, Royal Society Rooms at 7.30pm. Preceeding each meeting, at 7.00pm, there is a video "Making the Most of the BBC Micro".

FORWARD PROGRAMME:

The program at the moment for the October to December meetings is as follows:

14 October : Recent videos of Brian Cattermole's visit to England including BBC Computer shows and a look at Watford Electronics.

11 November : A demonstration of the Acorn Archimedes, the fastest personal computer ever built.

9 December : Youth night with games, games and more games. There will also be supper and Christmas goodies for our last meeting of the year.

There is also a teaching class on at our workshop at Wellington College every fourth Wednesday of the month. At the subject is "How to write BBC Basic". Later in the year there will be another course on "How to Write Machine Code". If you are interested, phone Don Churchill on 848-235 to be enrolled in these free courses.

~~~~~~
~~~~~~

LOCAL USER GROUP NEWS

HAWKES BAY

Mis-quote: 'I also serve who only makes the tea.'

We are all out here in Hawkes Bay, over fifty financial members, doing our different things with our Beebs. We now meet every second Sunday afternoon of the month at a convenient school; it's a real family party, mums, mostly dads and kids, grandchildren even, enough video games computer setups to keep all happy, and, as dad usually brings the kids, mums who stay at home can't gripe about him not digging the garden. It might not work as well in summer, but so far it's a great improvement on evening meetings.

The Beeb has brought me many new friends; I don't want to embarrass them by naming them, but they know who they are, and I cannot thank them enough. I hate to miss a meeting, and in spite of all this high-tech, I think I really am just a tea-lady at heart.

Ed: This item from Elizabeth Hill was accompanied by a interesting description of her set up at home (lovingly called her Honey-Beeb) which I will hold for next month so that there is room for her advertisement below.

~~~~~~

For Sale:

Brother CE 70 daisywheel electronic typewriter

With centronics parallel interface
ribbon cable for BBC computer connection.

This versatile dual outfit is ideal for printing out single sheets office-type work having a very high quality attractive type face.

As new, professionally serviced and maintained.

What offers for Brother and interface together?

Elizabeth Hill,
191 Flaxmere Ave., Hastings,
Ph: (070) 797541

~~~~~~

SELFTEST

by D S McDonald, Wellington.

"SELFTST" is a learning program. The user teaches the computer with sentences in the form "QUESTION?ANSWER." The computer reviews the sentences, when the user can correct any sentences that are wrong. The computer then asks the questions in random order or the order of input.

The candidate scores one mark for each character of the answer that is correct. If a character is wrong the computer beeps and prints the correct character. The time taken to answer is recorded. The results are candidate number and name, score, maximum possible score, percent, and time. The user can resit the test (R), teach new sentences (T or N), or quit (Q).

Sample data.

Books of the old and new testament,
Names of animals starting with A-Z,
English to French vocabulary, etc.

```
1 CH?CHRONICLES
  X?XANTHURA
  THE WINDOW?LA FENETRE
```

(Program can be modified to dimension for 65 sentences)

```
 10 REM Program SELFTST, Self-Test, by D.McDonald
 20 REM Version 2.0 , 23 July 1987
 30 NDIM=50
 40 DIM SENTENCE$(NDIM),PNTR(NDIM)
 50 MODE7:REM VDU14
 60 PROCTITLEINTRO
 70 PROCLEARN
 80 PROCREVIEWEDIT
 90 PROCTEST
100 PRINT'''"TYPE R - TO RESIT TEST"'''"T - TO TEACH
    COMPUTER NEW SENTENCES"'''"OR Q - TO QUIT."
110 REPEAT
120 A$=GET$
130 UNTIL INSTR("RTQ",A$)
140 ON INSTR("RTQ",A$) GOTO 90,60,150
150 PRINT'''"END."
160 END
```

.....*SELFTST* Contd >>

.....*SELFTST* Contd >>>

```
170 DEF PROCTITLEINTRO
180 VDU7
190 REM
200 RESTORE
210 DATA129,130,131,132,133,134,135
220 READ RED,GRE,YEL,BLU,MAG,CYA,WHI
230 REM
240 DATA  16,  136,137,  140,  141
250 READ GRAPHICS,FLASH,STEADY,NHEIGHT,DBLHEIGHT
260 REM
270 DATA  152,   153,   154, 156,  157,  158,  159
280 READ CONCEAL,CONTIG,SEPAR,BLKBK,NEWBK,HOLD,RELEASE
290 REM
300 DATA 8
310 READ LFT
320 REM
330 CLS
340 VDU BLU,NEWBK,RED,DBLHEIGHT:
    PRINT"S E L F - T E S T"
350 VDU BLU,NEWBK,RED,DBLHEIGHT:
    PRINT"S E L F - T E S T"
360 PRINT'''"SELFTST BY D.MCDONALD"''
    "VERSION 2.0 , 23 JULY 1987"''STRING$(40,"-")''
370 PRINT"TESTS PERSON(S) ON A TEXT QUIZ WHICH THE USER
    ENTERS."''''"MAY BE QUESTION AND ANSWER,
    MULTI-CHOICE, OR MEMORY."
380 PROCSPACECONTINUE(5,20)
390 ENDPROC
400 REM
410 DEF PROCSPACECONTINUE(COL,ROW)
420 PRINT TAB(COL,ROW);:VDU BLU:
    PRINT"PRESS SPACE BAR TO CONTINUE :"
430 PROCDELAY(1)
440 *FX 21,0
450 REPEAT UNTIL GET$=" "
460 CLS
470 ENDPROC
480 REM
490 DEF PROCDELAY(N)
500 FORDELAY=1TO N*2000:NEXTDELAY
510 ENDPROC
520 REM
530 DEF PROCLEARN
540 INPUT"ENTER SUBJECT OF TEST."''"EG. SPELLING,
    VOCABULARY, SONG, MEMORY, VERSE, ETC.",SUBJECT$
```

.....*SELFTST* Contd >>>

```
550 INPUT''"ENTER NAME",TYPIST$
560 REM KEYBOARD INPUT ONLY
570 CLS:VDU RED:PRINT"SELF-TEST  --";SUBJECT$'TYPIST$
580 PRINT'"ENTER MEMORY TEXT :-"''"ONE SENTENCE AT A
    TIME, MAX 255 CHARS (6LINES)."''"PRESS RETURN"'
590 PRINT"TYPE ? (OPTIONAL) AFTER PART OF SENTENCE
    WHICH IS TO BE GIVEN AS A CUE, OR QUESTION --
    FOLLOW-ON ANSWER."''"RETURN"'
600 PRINT"MAXIMUM "NDIM" SENTENCES."''
610 VDU RED:PRINT"ENTER NULL LINE (RETURN)":VDURED:
    PRINT"TO END INPUT OF MEMORY TEXT."'
620 PROCSPACECONTINUE(3,22)
630 REM
640 N=1
650 REPEAT
660 PROCSENTENCE
670 N=N+1
680 UNTIL N>NDIM OR SENTENCE$(N-1)=""
690 N%=N-2
700 PRINTTAB(0);N%" SENTENCES."
710 NAME$=""
720 CAND=0
730 SCORE=0
740 NSTART%=1:NFIN%=N%
750 FORI=NSTART%TONFIN%:PNTR(I)=I:NEXTI
760 PROCSPACECONTINUE(5,24)
770 ENDPROC
780 REM
790 DEF PROCSENTENCE
800 FOR L=1TO 7:PRINTTAB(0); L:NEXT L
810 PRINT TAB(0,VPOS-7);:VDURED:PRINTTAB(0)"SENTC.";N;
820 VDUWHI:INPUTLINE SENTENCE$(N)
830 PRINTSTRING$(240," "),TAB(0,VPOS-4)
840 ENDPROC
850 REM
860 DEF PROCREVIEWEDIT
870 CLS:VDURED:PRINT"REVIEW/EDIT"'''"PRESS SPACE BAR TO
    CONFIRM EACH","SENTENCE OF TEXT"'''"OR E - TO
    EDIT(TYPE IN AGAIN IF WRONG.)"'''"L - TO SKIP TO
    LAST."''
880 FORN=1TON%
890 PRINT''TAB(0);PNTR(N);SENTENCE$(PNTR(N))
900 REPEAT E$=GET$
910 UNTIL E$=" "ORE$="L"ORE$="E"
920 PROCDELAY(1):*FX21,0
```

.....*SELFTST* Contd >>>

.....*SELFTST* Contd >>>

```
 930 IFE$=" " THEN960
 940 IFE$="L"THENN=N%-1:GOTO960
 950 VDU7:PROCSENTENCE
 960 NEXTN
 970 PRINT''
 980 ENDPROC
 990 REM
1000 DEF PROCTEST
1010 CLS:VDURED:PRINT"TEST"
1020 PRINT''"NO. OF QUESTIONS "N%
1030 INPUT''"START TEST AT SENTENCE NO.",NSTART%
1040 IFNSTART%<1ORNSTART%>N%PRINT"WRONG NO.":GOTO1020
1050 INPUT''"FINISH TEST AT SENTENCE NO.",NFIN%
1060 IFNFIN%<NSTART%ORNFIN%>N% PRINT"WRONG NO.":
      GOTO1050
1070 FORI=NSTART%TONFIN%:PNTR(I)=I:NEXTI
1080 REPEAT
1090 PRINT''"TYPE SPACE - TAKE SENTENCES IN ORDER"''
      "OR    R    - RANDOM ORDER."
1100 O$=GET$
1110 UNTILO$=" " OR O$="R"
1120 IFO$="R" THEN PROCMIX
1130 PRINT''"LAST CANDIDATE ";CAND;
      NAME$'"..SCORE ="SCORE
1140 INPUTLINE'"SELECT TEST - ENTER CANDIDATE NAME"'
      "OR REVIEW - PRESS RETURN",NAME$
1150 REM HEADING
1160 CLS:
      PRINT"SUBJECT OF TEST -"SUBJECT$''"BY -"TYPIST$
1170 PRINT''"START SENTENCE ";NSTART%" FINISH "NFIN%
1180 IFO$="R" PRINT'"SENTENCES IN RANDOM ORDER."
1190 IF NAME$=""THEN PROCREVIEWEDIT:GOTO1290
1200 PRINT''"CANDIDATE -"NAME$''
      "PRESS SPACE TO START CLOCK."
1210 REPEAT UNTILGET$=" "
1220 HOLD=0
1230 CAND=CAND+1
1240 SCORE=0:MAXSCORE=0
1250 FORI=NSTART%TONFIN%
1260 PROCQUESTION
1270 NEXTI
1280 PROCRESULTS
1290 ENDPROC
```

.....*SELFTST* Contd >>>

.....*SELFTST* Contd >>>

```
 1300 REM
 1310 DEF PROCMIX
 1320 REM SORT SENTENCES IN RANDOM ORDER
 1330 FORI=NSTART%TONFIN%
 1340 J=(NSTART%-1)+RND(NFIN%-NSTART%+1)
 1350 TEMP=PNTR(I)
 1360 PNTR(I)=PNTR(J)
 1370 PNTR(J)=TEMP
 1380 NEXTI
 1390 ENDPROC
 1400 REM
 1410 DEF PROCQUESTION
 1420 S$=SENTENCE$(PNTR(I))
 1430 CLS
 1440 TIME=HOLD
 1450 X=INSTR(S$,"?")
 1460 IFX THENPRINTLEFT$(S$,X);
 1470 MAX=LEN(S$)-X
 1480 CHARS=0
 1490 FORX=X+1TOLEN(S$)
 1500 A$=GET$
 1510 IFA$=MID$(S$,X,1) PRINTA$;:CHARS=CHARS+1:GOTO1530
 1520 VDU7,RED,NEWBK,WHI:PRINTA$;:VDUBLKBK:
      PRINTMID$(S$,X,1);
 1530 NEXTX
 1540 HOLD=TIME
 1550 SOUND1,-15,CHARS,20
 1560 PRINT'''"CORRECT ";CHARS;" MAX. ";MAX;" %= ";
      INT(CHARS/MAX*100+.5);" TIME ";INT(HOLD*.01)'''
 1570 IFMAX-CHARS>6 VDURED:PRINT"    000 000"
 1580 SCORE=SCORE+CHARS
 1590 MAXSCORE=MAXSCORE+MAX
 1600 PROCSPACECONTINUE(5,15)
 1610 ENDPROC
 1620 REM
 1630 DEF PROCRESULTS
 1640 CLS:PRINT"RESULTS..."'''"CANDIDATE. ";NAME$,CAND
 1650 PRINT"NO. OF QUESTIONS ",NFIN%-NSTART%+1
 1660 PRINT"SCORE",SCORE
 1670 PRINT"MAXSCORE. ",MAXSCORE
 1680 PRINT"PERCENT=",.1*INT(SCORE*1000/MAXSCORE+.5)
 1690 PRINT"TIME ",HOLD*.01
 1700 ENDPROC
```

~~~~~~
~~~~~~

SERVICE AGENTS AND DEALERS

The New Zealand network of service and dealers for the Acorn range of computers is as follows:

AUCKLAND:

	Telephone
Sales and Service- Barson Computers NZ Ltd, PO Box 26-287 1 Ngaire Avenue, Auckland 3	504-049 504-631
Other Dealers- Computer Terminal 257 Hinemoa St, B'head.	419-0543
Database Services PO Box 94 8 Kowhatu Rd	665-540

GORE:

	Telephone
Lyndon McEntee, Educomp, PO Box 255	7365

HASTINGS:

	Telephone
Whitcoulls Ltd 113 East Heretaunga St.	84-394

WELLINGTON:

	Telephone
Service- Data Group Limited PO Box 1779 2nd Flr, Construction House 80-82 Kent Tce	857-974
Dealer- Micro Interface (NZ) Ltd PO Box 14-441 Kilbirnie	882-444

1.8.87

FOR SALE:

Solidisk 32k Sideways Ram (includes software on disc for 16k printer buffer, DFS with page at &E00, and Ramdisc)	$50
Solidisk DFS 2.2M Rom (compatible with 8271 and 1770 disk controllers)	$45

Steven Webb, Phone (04) 648 048
10 Wise Street, Wainuiomata.

MEMBERSHIP

Membership of the National User Group is on payment of an annual subscription from April to March, which is $20.00 for the 1987/8 year. A local branch levy may be payable which entitles attendance at meetings as detailed on the back page. Joining or renewing members get all the back issues of *Beeblet* for the current year - those joining after 1 October may have half the annual subscription waived and only receive *Beeblet* from 1 October.

USER GROUP MEETINGS

AUCKLAND: -2nd and 3rd Wednesday of each month at Social Club, Auckland Hospital, Grafton Road at 7.30pm. Contact John Buchanan (Secy), 13 Ohinerau St, Remuera; David Wilkin (Pres), 29 Rose Camp Rd, Birkdale

North Shore: -last Wednesday of the month, 7.30pm, Salvation Army Hall, Glenfield Road, Glenfield. Contact: Chris, Phone 419-0543 (wk)

CHRISTCHURCH: -fortnightly workshops, Monday 6.30pm at Hagley High School. phone Michael 582-267.

DUNEDIN: -first Thursday of the month. Contact: Martin McDowall, Phone 42-031

HAWKES BAY: -Alternative months Hastings and Napier. Contact Kendall: Napier 435-624, Major: Napier 447-729, Mitch: Hastings 85-395.

INVERCARGILL: -Phone Evan 330-444 Invercargill.

NELSON: -second Monday of the month, 7.30pm, Nelson Baptist Church, Bridge St. Contact: Peter, Richmond 8975; Ken, Nelson 85-150; Richard, Richmond 8489.

PALMERSTON NORTH:-Phone Brian Knowles 79-319

ROTORUA: -first Tuesday of the month at Rotorua High School. Phone: Ian 85-583, Joan 81-536 or Tony 89-316.

TAUPO: -every second Wednesday, 7.00pm. Contact: David 84-215 (h), 85-124 (wk).

TAURANGA: -every second Wednesday. ph. Chris & Jane 65-076.

TIMARU: -Phone Lloyd van der Krogt Timaru 61-412.

TOKOROA: -1st Friday of month, 7.30pm, Tokoroa High School. Contacts: Don Fraser, 68-813 or Graeme Robertson, 65-346 Tokoroa.

WAIKATO: -first Monday of the month, 7.30pm, Waikato Tech Institute. Ph. Alison or John, Morrinsville 6695.

WELLINGTON: -second Wednesday of the month, 7.30pm, Royal Society Lecture Theatre, Turnbull St, Thorndon.
-Workshops: fourth Wednesday of month, other details: ph. Don 848-235 or Mike 785-437.

OTHER CENTRES: -let us know and we will publish details here.

MEMBERSHIP

For membership details, see previous page.

BEEBLET

PUBLISHED: monthly except January and mailed to financial members.

CONTRIBUTIONS: -most welcome. Listings should be sent in on tape or disc. (Tape - 2 copies, one at 300 baud please). Please include written explanation of listing; text and graphics only material welcome too.

DEADLINES: -For a particular issue, the last day for material is the fifteenth day of the month prior to publication.

ADVERTISING: -No fixed charge is made for advertising. Supply camera ready, deadlines as above.

Copyright (c) BBC/ACORN COMPUTER USER GROUP OF NZ INC.
Registered Publication with NZPost

THE BBC/ACORN COMPUTER USER GROUP OF NZ INC, PO Box 9592, WELLINGTON

Reproduced by SANDIFORD PRINTERS